

Bespreking Examen Automaten en berekenbaarheid

11 juni 2018

Prof. Dr. Bart Bogaerts

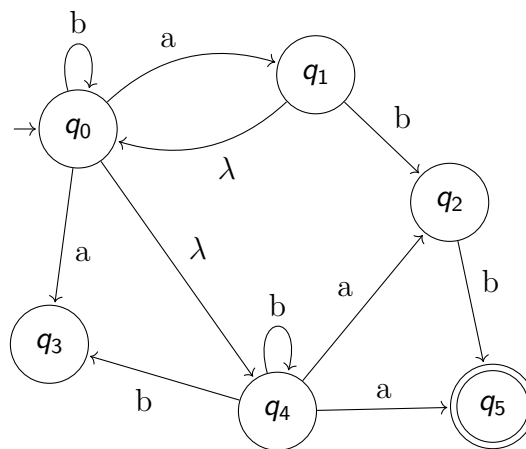
15 mei 2019

Sommige vragen worden hieronder opgelost; andere niet. In *italic* vindt je besprekingen die *niet* tot mijn oplossing behoren, maar die extra uitleg zijn.

Vraag 1 (3pt)

Zet onderstaande niet-deterministische FSA (finite state automaton) om naar een **mini-male** deterministische FSA, door gebruik te maken van de geziene procedures. Teken het resultaat.

Toon aan dat je automaat minimaal is.



Oplossing We zetten deze automaat eerst om naar een DFA.

als je wil, kan je op dit moment eerst ee nnutteloze state (q_3 verwijderen). Die state zal ons nooit helpen. Als je dat doet, zeg dan wel expliciet waarom!

Intuitief bestaat elke state van de DFA uit de verzameling states die bereikbaar zijn vanuit de start-state van de NFA na het volgen van een bepaalde string. De start-state van de DFA bevat dan ook twee NFA states, namelijk q_0 en q_4 , i.e., de start-state is $c_\lambda(q_0) = \{q_0, q_4\}$. Voor het opstellen van de minimale DFA zal ik gebruik maken van

volgende tabel:

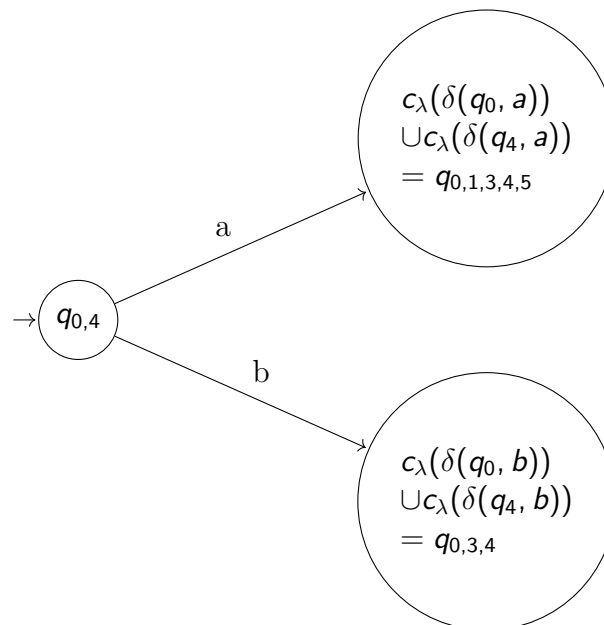
	$c_\lambda(\delta(\cdot, a))$	$c_\lambda(\delta(\cdot, b))$
q_0	$\{q_0, q_1, q_3, q_4\}$	$\{q_0, q_4\}$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\emptyset$	$\{q_5\}$
q_3	$\emptyset$	$\emptyset$
q_4	$\{q_2, q_5\}$	$\{q_3, q_4\}$
q_5	$\emptyset$	$\emptyset$

Merk op dat deze tabel een adnere is dan in sommige oplossingen oefenzittingen! Ik vind het eenvoudiger om per state te werken ipv ineens met verzamelingen. De reden hiervoor: er zijn maar 6 states maar mogelijks heel veel verzamelingen states...

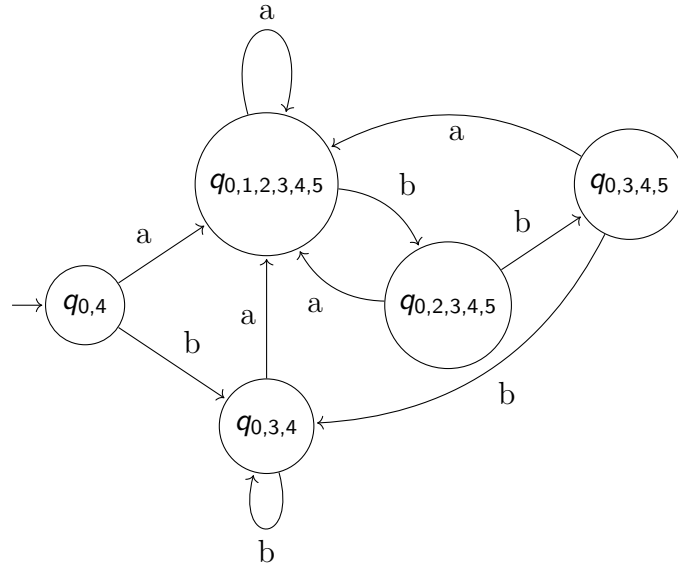
Dus: de intiële state bevat q_0 en q_4 , genoteerd zoals hieronder:



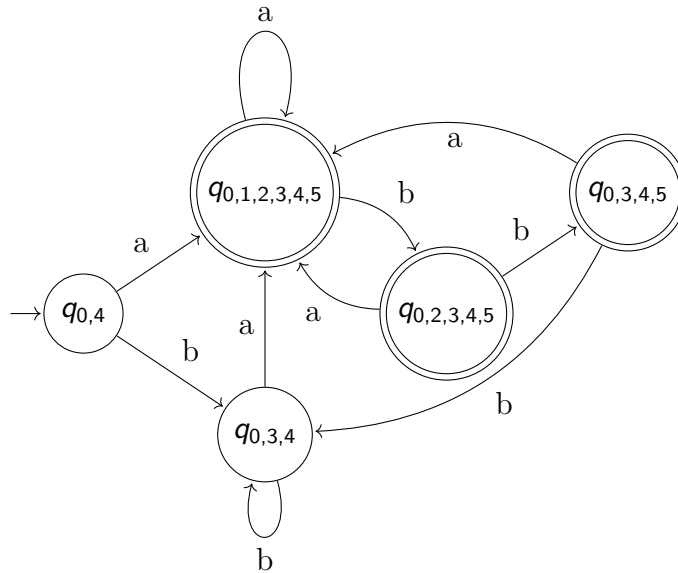
We vullen deze verder aan, bijv met:



We blijven dit proces herhalen waarbij we zorgen dat er een uitgaande transitie voor a en b uit elke state is en de uitganade state berekend wordt mbv de tabel.



Alle states die een originele accepterende state bevatten worden accepterend.



De volgende stap is deze DFA minimaliseren. Initieel hebben we de volgende partitie (finale vs niet-finale states) van states (waarbij in een ander element van de partitie zitten betekent: distinguishable zijn).

$$\{\{q_{0,4}, q_{0,3,4}\}, \{q_{0,1,2,3,4,5}, q_{0,2,3,4,5}, q_{0,3,4,5}\}\}$$

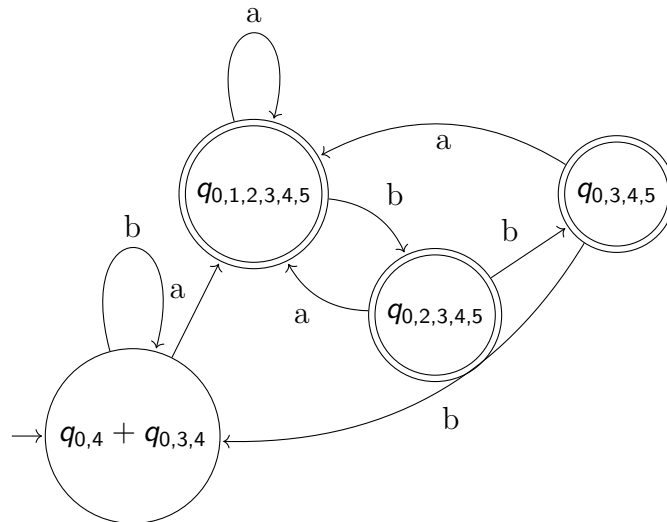
Aangezien $\delta(q_{0,3,4,5}, b)$ niet accepterend is en voor de andere states uit zijn partitie b volgen wel acceptered krijgen we:

$$\{\{q_{0,4}, q_{0,3,4}\}, \{q_{0,1,2,3,4,5}, q_{0,2,3,4,5}\}, \{q_{0,3,4,5}\}\}$$

Maar: opnieuw: via b volgen zien we dat $\delta(q_{0,1,2,3,4,5}, b)$ en $\delta(q_{0,2,3,4,5}, b)$ uit een verschillende partitie komen, dus deze twee states zijn distinguishable. Dit geeft:

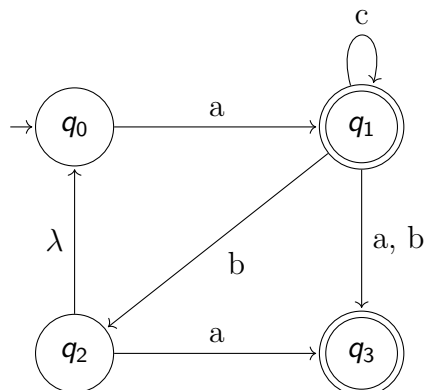
$$\{\{q_{0,4}, q_{0,3,4}\}, \{q_{0,1,2,3,4,5}\}, \{q_{0,2,3,4,5}\}, \{q_{0,3,4,5}\}\}$$

Nu geldt dat $\delta(q_{0,4}, a) = \delta(q_{0,3,4}, a)$ en $\delta(q_{0,4}, b) = \delta(q_{0,3,4}, b)$ dus deze states zijn indistinguishable. Het uiteindelijke resultaat (de geminimaliseerde dfa) is:



Vraag 2 (2pt)

Geef de reguliere expressie van de taal die wordt geaccepteerd door deze automaat. Volg de geziene procedure, die gebruik maakt van GTGs (General Transition Graph).



Nog geen commentaar/oplossing.

Vraag 3 (2pt)

Is de volgende taal regulier? Indien ja, geef een DFA voor de taal, anders bewijs met de pompstelling.

$$L_1 = \{0^i 1^j \mid 0 \leq 2i \leq j\}$$

Nog geen commentaar/oplossing.

Vraag 4 (3pt)

Zet volgende grammatica in Chomsky Normal Form:

$$S \rightarrow aba|aC|CD|D$$

$$A \rightarrow aA|bB|\lambda$$

$$B \rightarrow b|Aa$$

$$C \rightarrow aC|D|\lambda$$

$$D \rightarrow bb|Sb|\lambda$$

Opmerking: Technisch gezien is er iets mis met deze vraag. De definitie die wij gebruiken (die uit het boek) garandeert enkel het bestaan van chomsky normaalvorm indien $\lambda \notin L(G)$. Dat is hier niet het geval. Een alternatieve definitie van chomsky normaalvorm laat ook toe dat er een productie $S \rightarrow \lambda$ is (waarbij S de startvariabele is). Om deze vraag volledig op te lossen, zal je dat nodig hebben!

Vraag 5 (6pt)

Zijn volgende talen context-vrij? Als de taal context-vrij is, geef dan een niet-deterministische push-down automaton. Als de taal niet context-vrij is, bewijs dit aan de hand van de pompstelling.

a) $L_1 = \{(a+b)^n(a+c)^n | n \geq 0\}$

b) $L_2 = \{a^k b^n c^m | m = kn \text{ en } k, n, m > 0\}$

Nog geen commentaar/oplossing.

Vraag 6 (4pt)

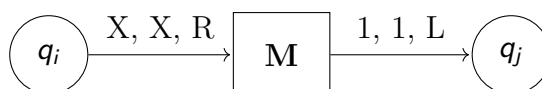
Gegeven is een single-tape Turing Machine $\mathbf{M} (Q_M, \{1, \bullet\}, \Gamma_M, \delta_M, q_{M_0}, \square, \{q_{M_f}\})$, die het product xy in unaire representatie berekent. Deze machine verwacht een aantal '1' symbolen (x) op de tape, gevolgd door het ' $\bullet$ ' symbool, gevolgd door een aantal '1' symbolen (y), en schrijft als output x maal y op de tape. De output start op dezelfde positie als de oorspronkelijke input. De head wordt terug op het begin van de output geplaatst.

	...	0	1	2	3	4	5	6	7	...
Voor:	$\square$	$\square$	1	1	$\bullet$	1	1	1	$\square$	$\square$
				Δ						
	...	0	1	2	3	4	5	6	7	...
Na:	$\square$	$\square$	1	1	1	1	1	1	$\square$	$\square$
				Δ						

Maak gebruik van $\mathbf{M}$ om een **single**-tape Turing Machine $\mathbf{P}$ te maken, die x^y in unaire representatie berekent, met $x, y > 0$. De output start op dezelfde positie als de oorspronkelijke input. De head mag op een willekeurige positie eindigen.

	...	0	1	2	3	4	5	6	7	8	9	...
Voor:	$\square$	$\square$	1	1	$\wedge$	1	1	1	$\square$	$\square$	$\square$	$\square$
				Δ								
	...	0	1	2	3	4	5	6	7	8	9	...
Na:	$\square$	$\square$	1	1	1	1	1	1	1	1	$\square$	$\square$
				Δ								

Maak gebruik van $\mathbf{M}$ in $\mathbf{P}$, als aangeduid op het volgende voorbeeld:



Geef uitgebreide uitleg over de werking van je Turing Machine. Zonder uitleg krijg je geen punten op deze vraag!

Nog geen commentaar/oplossing.